Competition Alleviates Present Bias In Task Completion

Aditya Saraf

June 19, 2020

Contents

1 Introduction **1**
- 1.1 Prior Work 3
- 1.2 Our Model 4
- 1.3 Summary of Results 6

2 Nash Equilibria Without Bias **6**

3 Nash Equilibria to Elicit Optimal Behavior from Biased Agents **8**
- 3.1 Nash Equilibria on the n-fan 8
- 3.2 Nash Equilibria in General Graphs 10

4 Bayes-Nash Equilibria with Bias Uncertainty **12**
- 4.1 Bayes-Nash Equilibria on the n-fan 12
- 4.2 Difficulties Finding Bayes-Nash Equilibria in Arbitrary Graphs 15

5 Conclusion and Future Directions **16**

A Proofs **17**
- A.1 Proof of Lemma 1 17
- A.2 Proof of Lemma 2 17
- A.3 Proof of Lemma 3 18
- A.4 Proof of Claim 2 18

1 Introduction

One of the most influential lines of recent economic research has been *behavioral* game theory. The majority of economics research makes several idealized assumptions about the behavior of rational agents to prove mathematical results. Behavioral game theorists question these assumptions and propose models of agent behavior that more closely align with human behavior. Through experimental research, behavioral economists have observed and codified several common types of cognitive biases, from loss aversion (the tendency to prefer avoiding loss to acquiring equivalent gains) to the sunk cost fallacy (the tendency to factor in previous costs when determining the best future course of action) to present bias (the current topic). One of the primary purposes of many mathematical theorems in game theory is to offer predictive power, which is especially important in the many computer science applications of these results, from modern ad auctions to cryptocurrency protocols. But when we'd like to predict human behavior, it makes sense to start with mathematical models that include observed human biases. Thus, rather than viewing behavioral game theory as conflicting with the standard mathematical approach, the experimental results of behavioral game theory ought to feed back into more sophisticated mathematical models. This thesis builds upon recent work by Kleinberg et al., who investigate a mathematical model for planning problems where agents are present biased [KO14, KOR16, KOR17].

Present bias is the tendency to prefer to minimize current loss over future losses. This is a ubiquitous bias in human behavior that can model diverse phenomena. The most natural example is procrastination, the familiar desire to delay difficult work, even when this predictably leads to bigger problems later. Present bias can also model the tendency of firms to prefer immediate gains to long-term gains and the tendency of politicians to prefer immediate results to long-term plans. The simplest model of present bias is to multiply costs in the current time period by present bias parameter b when making plans. Thus, the model can be seen as a special case of hyperbolic discounting, where costs are discounted in proportion to how much later one would experience them. But even this simple model induces time-inconsistency that suffices to model a rich set of human behavior.

Examples of time inconsistent behavior extend beyond procrastination. For example, one might undertake a project, and abandon it partway through, despite the underlying cost structure remaining unchanged. One might fail to complete a course with no deadlines, but pass the same course with weekly deadlines. One might pay for a gym membership, but fail to use it. Kleinberg et al. presented the key insight that this diverse range of phenomena can all be expressed in a single graph-theoretic framework [KO14].

Consider a directed, acyclic graph G, with designated source s and sink t. We refer to these graphs as task graphs, with s being the start of the task and t the end. A path through this graph is a particular plan to complete the task; each edge represents one step of the plan. The graph is weighted, and the goal of any agent is to complete the task as efficiently as possible, which corresponds to taking the cheapest path through the graph. With an optimal agent, this problem is trivial. A *naive* present biased agent with bias parameter b behaves as follows. At s, they evaluate each path to t by multiplying the first edge cost on the path by b, and choosing the cheapest path with this metric. They then take *one* step along this path, say to v, and re-evaluate all the paths from v to t. Notice that they may deviate from the path they planned to take at s, as now the edges leaving v are multiplied by b. This is the sense in which the agent is naive. They do not correctly reason about how their future self will behave, and thus exhibit time-inconsistent behavior. See Figure 1 for an example.

The power of this graph theoretic model is that it allows us to answer questions that quantify over a large set of planning problems, and to formally investigate which tasks present the "worst-

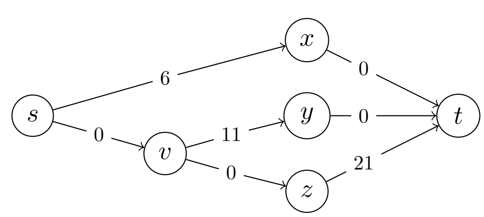


Figure 1: The optimal path is (s, x, t) with total cost 6. However, an agent with bias $b = 2$ will take path (s, v, z, t), with cost 21. Importantly, when the agent is deciding which vertex to move to from s, they evaluate x as having total cost 12, while v has total cost 11. This is because they assume they will behave optimally at v by taking path (v, y, t). However, they apply the same bias at v and deviate to the worst possible path.

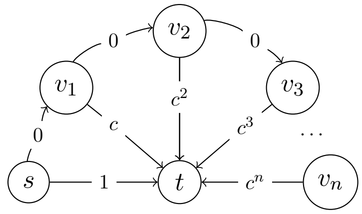


Figure 2: A naive agent with bias $b > c$ will continually choose to delay finishing the task.

case" for procrastination. This is useful both to understand how present-biased behavior differs from optimal behavior and to design tasks to accommodate present bias. We now briefly summarize the existing literature, to motivate why we introduced competition to this model.

1.1 Prior Work

The most striking result is that there are graphs where the *cost ratio* (the ratio of the optimal agent's cost to the biased agent's cost) is exponential in the size of the graph. In addition, all graphs with exponential cost ratio have a shared structure – they all have a large k-fan as a graph minor (and graphs without exponential cost ratio do not) [KO14, TTW+17]. So this structure encodes the worst-case behavior for present bias in the standard model (and we later show how competition is especially effective in this graph). An n-fan is pictured in Figure 2.

The exponential cost ratio demonstrates the severe harm caused by present bias. This naturally begs the question of how task designers ought to deal with present bias. [KO14] propose a model where a reward is given after finishing the task, and where the agent will abandon the task if at any point, they perceive the remaining cost to be higher than the reward. Unlike an optimal agent, a biased agent may abandon a task partway through. Figure 3 uses the gym membership example from earlier to show this. As a result, they give the task designer the power to arbitrarily delete vertices and edges, which can model deadlines, as Figure 4 shows. They then investigate the structure of *minimally motivating subgraphs* – the smallest subgraph where the agent completes the task, for some fixed reward. Follow-up work shows that finding *any* motivating subgraph is NP-hard [TTW+17]. Instead of deleting edges, [AK19] consider the problem of spreading a fixed reward onto arbitrary vertices to motivate an agent, and find that this too is NP-hard (with a constrained budget).

The above results all focus on *accommodating* present bias rather than *alleviating* it. By that, we mean that the approaches all focus on whether the agent can be convinced to complete the task – via edge deletion or reward dispersal – but not on guarding the agent from suboptimal behavior induced by their bias. [KOR16] partially investigates the latter question in a different model. They study *sophisticated* present-bias, where agents plan around their present bias. They also flip the model into a reward setting – edge costs now represent rewards, and the goal is to collect as much reward as possible while going from s to t. Sophisticated agents can do exponentially poorly here,

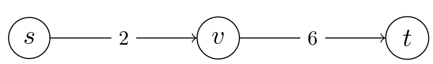


Figure 3: Let (s, v) represent buying a gym membership and (v, t) represent working out regularly for a month. At t, the agent receives a reward of 11 due to health benefits. With bias $b = 2$, the agent initially believes this task is worth completing, but due to his bias, abandons the task at vertex v, after having already purchased the membership. This same example works more generally for project abandonment, where (s, v) could represent the easier planning/conceptual stage while (v, t) represents the more difficult execution.

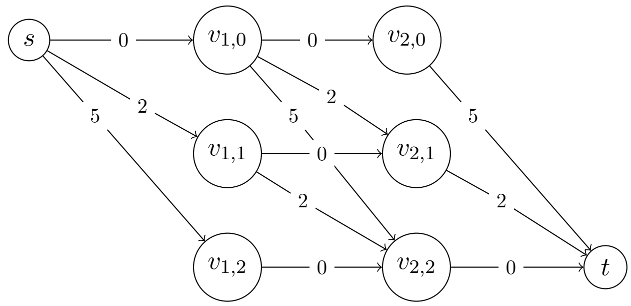


Figure 4: This example represents a three week course with two assignments. Let $v_{i,j}$ correspond to being in week i with j assignments completed, and suppose the reward of the course is 9. Note that a student with bias $b = 2$ will procrastinate until $v_{2,0}$ and then ultimately abandon the course. However, if the instructor required the first assignment to be completed by week 2 (which corresponds to removing $v_{2,0}$), the student would complete the course.

despite behaving nearly optimally in the cost setting. To resolve this, they consider several types of *commitment devices* – tools by which sophisticated agents can constrain their future selves. Namely, they separately investigate three schemes:

1. Allowing agents to modify the instance by paying upfront to distribute an equivalent reward on arbitrary edges. Note that the agent's present bias amplifies the upfront cost. For example, one might open an especially inconvenient retirement account to make early withdrawals more difficult. Even with this power, the reward ratio remains exponential in some instances.

2. Allowing agents to add arbitrary zero-reward edges. This eliminates the exponential worst case; the new worst case instances see the biased agent accruing $1/bn$ fraction of the optimal reward.

3. Allowing agents to delete arbitrary edges. This is supposed to model (self-imposed) deadlines. They show that, for any graph, deleting an ε fraction of its edges brings the reward ratio up to $\Omega(n^{-\varepsilon/2})$.

There are several limitations with this approach that we hope to address. Their first model seems reasonable, but doesn't allow agents to beat the exponential worst case. The second model seems unjustified; adding arbitrary zero-reward edges doesn't seem to model a realistic commitment device. Further, there's no clear analog of this in the cost setting (infinite cost edges would never be taken). Their third model suffers from the same issue (but less severely). While some edge removals can be construed as deadlines, it's not clear that arbitrary edge removals always correspond to deadlines. Finally, while commitment devices can apply to sophisticated agents, who anticipate their bias, they don't make sense for naive agents.

In addition to these details, these commitment devices don't fulfill our larger goal. We want a model that simultaneously explains why present-biased agents may not perform exponentially poorly in "natural" games and instructs task designers on how to encourage biased agents towards optimal behavior. We now describe our model, which adds a game theoretic element in the form of competition to accomplish these goals.

1.2 Our Model

In our model, tasks are still represented as a directed, acyclic graph G, with designated source s and sink t. There are two naive present-biased agents, A_1 and A_2, both with bias b, who are competing to get to t first. The cost of a path is the sum of the weights along the path, and time is represented by the number of edges in the path. In other words, each edge represents one unit of time. The first agent to get to t gets a reward of r; ties are resolved by evenly splitting the reward. To see how the agents behave, recall that naive agents believe that they will behave optimally in the future. Thus, when the agent standing at u is evaluating the cost of going to v, they add $bc(u,v)$ to the cost of the optimal path from v to t minus the (expected) reward of that path. More formally, let $\mathcal{P}(v \to t)$ denote the set of paths from $v \to t$ and let $P(s \to u)$ denote the path the agent has taken to u. If we let $S_n(u)$ denote the successor of node u for a naive agent with bias b:

$$S_n(u) = \operatorname*{argmin}_{v:(u,v)\in E} b \cdot c(u,v) + \min_{P(v\to t)\in\mathcal{P}(v\to t)} c(P(v \to t)) - \mathbb{E}[R(P(s \to u) \cup (u,v) \cup P(v \to t))]$$

Where $c(P) = \sum_{e\in P} c(e)$ denotes the cost of path P and $\mathbb{E}[R(P(s \to u) \cup (u,v) \cup P(v \to t))]$ denotes the expected reward of taking path P from $v \to t$ given that one has taken path $P(s \to u)$ from $s \to u$ and will take edge (u,v). This expectation is based on what A_1 knows about which path A_2 will take (i.e., if we're searching for a Nash equilibrium on path P, then A_1 assumes A_2 will take P). Though the successor equation looks complicated, note that the optimal agent chooses successors in exactly the same way, except for the bias parameter b. See Figure 5 for an example.

As we stated earlier, this model of competition can both explain the outcomes of natural games and inform task designers on how to elicit optimal behavior from biased agents. We provide motivating examples of both. For a natural game, consider the classic example of two companies competing to expand into a new market. Both companies want to launch a similar product, and are thus considering the same task graph G. The companies are also present biased, since shareholders often prefer immediate profit maximization/loss minimization over long term optimal behavior. The first company to enter the market gains an insurmountable advantage, represented by reward r. If the companies both enter the market at the same time, they split the market share, each getting reward $r/2$. What equilibria do we expect to see in this game?

For a designed game, consider the problem of encouraging students to submit final projects before they are due. As an instructor, you set a deadline near the end of finals week so that students can have the flexibility to complete the project when it best fits their schedule. But you are also aware that (1) students tend to procrastinate and (2) trying to complete the final project in a few days is much more challenging than spreading it out. You would thus like to convince students to submit their assignments early, *without* changing the deadline (to allow flexibility for the students whom it suits best). One possible solution would be give a small amount of extra credit to the first submission. How might we set this reward to encourage early submissions?

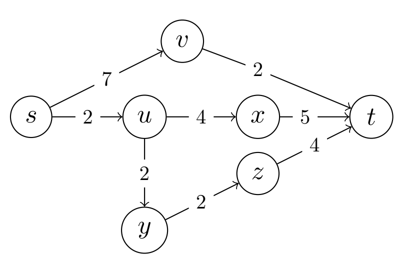


Figure 5: Suppose $r = 5$, the bias $b = 2$, and assume A_2 takes path (s,u,x,t). Then at s, A_1 prefers to take u for perceived cost $4 + 4 + 5 - 2.5 = 10.5$. Notice that, due to the reward, the path A_1 believes he will take from u is (u,x,t), despite (u,y,z,t) having lower cost. However, at u, A_1 evaluates the lower path to be cheaper, despite losing the race. This shows that a reward of 5 does not ensure a Nash equilibrium on (s,u,x,t) when $b = 2$.

For another example of a designed game, consider a gym that enjoys increased enrollment at the start of the year. However, their new customers tend to not visit the gym frequently, and eventually cancel their membership. To remedy this, they try a new promotion, where new customers are offered free fitness classes, and the first customer(s) to attend 5 such classes gets a discount on their annual membership. How effective might such a scheme be at encouraging their new customers to regularly use their gym?

The intuition in these three examples is that the competitive element of racing other agents will alleviate the harms of present bias. We now summarize to what extent this is true.

1.3 Summary of Results

Our model is interesting to analyze even without bias, so we first classify all Nash equilibria for an arbitrary task graph with unbiased agents. This analysis is straightforward after characterizing all dominated paths. Then, we introduce present bias. We find that a *constant*[1] reward induces a Nash equilibrium on the optimal path, for any graph. This is a substantial improvement over the exponential worst case experienced without competition. We then generalize to a setting where agents' biases are drawn iid from distribution B and, for the n-fan, describe the relationship between B and the reward required for a Nash equilibrium on the optimal path. For a wide range of distributions, we show that it's cheap to get optimal behavior with high probability. For the stronger goal of ensuring a constant expected cost ratio, it suffices to offer reward linear in n when B is not heavy-tailed; competition thus helps here as well. We then describe difficulties with finding Bayes-Nash equilibria in arbitrary graphs.

2 Nash Equilibria Without Bias

To build intuition, we describe the equilibria with unbiased agents, and how the analysis changes with the introduction of bias.

The simplest version of the problem assumes that both agents are optimal (i.e. not present biased). Notice that each path P in the graph is a strategy, with payoffs $u_w = r - c(P)$ and $u_l = -c(P)$, which correspond to winning and losing respectively (depending on the path the opponent takes). We first rule out dominated paths. Notice that if $u_l(P) > u_w(P')$, it's never beneficial to take path P'. Also, if $u_w(P) \geq u_w(P')$ and $|P| \leq |P'|$ (where $|P|$ is the number of edges in P), then P' is dominated. So, for any length k, the cheapest path of length k will dominate all other paths of length k. We also define two important paths: the *quickest* path is the single remaining path with minimum length, and the *cheapest* path is just the lowest cost/weight path. Note that the optimal path in the non-competitive setting is the cheapest path. Summarizing what we know about the remaining (non-dominated) paths:

1. There is at most one path of any length.
2. *Winning is better than losing*: for any pair of paths (P, P'), we know that $u_w(P) > u_l(P')$.
3. The quickest path has a reward at most r lower than the cheapest path (otherwise, losing with the cheapest path would be better than winning with the quickest path).
4. *Longer paths are more rewarding*: for any pair of paths, the one with higher reward (i.e. lower cost) will be longer. This also implies that the cheapest path is the longest remaining path.

[1] There is a mild dependence on the costs within the graph, but this is confined to one edge on the cheapest path.

We're interested in characterizing, across all possible task graphs, the (pure) Nash equilibria that occur in this problem.

For completeness, we first discuss two simpler models, in which ties result in players both winning or both losing. First, consider the version of the problem where ties give both players the reward.

Proposition 1. *Supposes ties are resolved by giving both players the reward. Then, for any task graph G, any non-dominated path is a (symmetric) pure Nash equilibrium. There are no other Nash equilibria.*

Proof. Index the paths by length as $P_1, \ldots, P_n$, where P_1 is the quickest path and P_n is the cheapest path. If one agent plays P_i, since tying is equivalent to winning, playing $P_{\leq i}$ gets a win and playing $P_{>i}$ guarantees a loss. Since winning is better than losing, one would never play $P_{>i}$. Since longer paths are more rewarding, one would play the longest path left, which is P_i. So any path is a pure Nash equilibrium and there are no other pure Nash equilibria. The best equilibrium for both players is to play the cheapest path. $\square$

Now, consider the version where ties give both players the lesser reward. This is a bit more complex.

Proposition 2. *Suppose ties are resolved by giving both players the lesser reward. Then, for any task graph G, there are pure Nash equilibria if and only if there are at most two non-dominated paths. In the trivial case where the quickest and cheapest path coincide, the pure Nash is for both players to play this path. Otherwise, the pure Nash is for one player to play the quickest and the other to play the cheapest. In all other cases, there are no pure Nash equilibria.*

Proof. In the case where the quickest path is the cheapest path, it's clearly a dominant strategy (and thus a Nash equilibrium) to play this path. So, suppose there are exactly two non-dominated paths remaining. Denote the quickest as Q and the cheapest as S. If one agent plays Q, the other can either get $u_l(Q)$ by playing Q, or $u_l(S)$ by playing S. Since the latter is higher, the other agent prefers to play S. Similarly, if one agent plays S, then playing Q yields $u_w(Q)$ while playing S yields $u_l(S)$. Since winning is better than losing, they will prefer Q. So (Q, S) and (S, Q) are the only pure Nash equilibria with two paths.

For the general case, suppose there are at least three paths. Index the paths by length as $P_1, \ldots, P_n$, where P_1 is the quickest path and P_n is the cheapest. The best response to playing P_1 is playing P_n, because tying is equivalent to losing and P_n is the cheapest. But the best response to P_i where $i > 1$ is P_{i-1}, since that's the longest winning path and thus the most rewarding. There are thus no pure Nash equilibria here. $\square$

Now, suppose we are in the original model, where the reward is evenly split on a tie. This case is more nuanced; some task graphs have different equilibria than others.

Proposition 3. *Suppose ties are resolved by giving both players $\frac{r}{2}$. Let task graph G be arbitrary. Let $P_1, \ldots, P_n$ be the paths in increasing order of reward, so P_1 is the quickest and P_n is the cheapest. Then, path P_i, where $i > 1$ is a symmetric Nash if and only if $u_t(P_i) \geq u_w(P_{i-1})$, where $u_t(P_i)$, the utility from tying on P_i, is $r/2 - c(P_i)$. P_1 is a symmetric Nash if and only if $u_t(P_1) > u_l(P_n)$. There are no other pure Nash equilibria.*

Proof. First, suppose the opponent picks P_i, where $i > 1$. If we play P_{i-1}, we get $u_w(P_{i-1})$. Since winning is greater than losing, we can drop all $P_{>i}$ from consideration. Similarly, since longer paths are more valuable we can ignore $P_{<i-1}$. Now, the only choices are between tying on P_i or winning

on P_{i-1}, so P_i is clearly a symmetric Nash exactly when $u_t(P_i) > u_w(P_{i-1})$. Similarly, if your opponent plays P_1, your options are to either tie on P_1 or lose; in the latter case, it's obviously best to lose on the cheapest path P_n. We thus get a symmetric Nash exactly when tying on P_1 is better than losing on P_n.

Now, if for all $i > 1$, $u_t(P_i) < u_w(P_{i-1})$ and $u_t(P_1) < u_l(P_n)$, we get the same result as when tying is equivalent to losing. The best response to playing P_1 is playing P_n. But the best response to P_i where $i > 1$ is P_{i-1}, due to the assumed inequalities. □

We now explore the biased version of this problem. For the unbiased version of the problem, we could take a "global" view of the graph, and think about paths purely in terms of their overall length and cost. But when agents are biased, the actual structure of the path is very important; time-inconsistency means that agents look at paths *locally*, not globally. It is thus very difficult to cleanly rule out dominated paths – even paths with exponentially high cost may be taken, as we explore next.

3 Nash Equilibria to Elicit Optimal Behavior from Biased Agents

We now assume that the agents are both naive, present biased agents, with shared bias parameter b. We characterize the Nash equilibria, first on the n-fan, and then for arbitrary graphs.

3.1 Nash Equilibria on the n-fan

We focus first on the n-fan since all graphs with exponential cost ratio for naive agents have a large k-fan as a graph minor [KO14]. For an example of the n-fan, refer back to Figure 2. We show that, with a very small amount of reward, competition encourages optimal behavior in naive agents. To aid our proofs and discussions regarding the n-fan, we define P_i as the path from s to t containing edge (v_i, t). Let P_0 correspond to the direct path (s, t).

Theorem 1. *Let G be an n-fan, and for any bias $b > c$, define $\varepsilon = b - c$. Then, with a reward of $r \geq 2\varepsilon$, there will be a Nash equilibrium on the optimal path, for two agents with bias b. Further, if $r \leq 2\varepsilon c^{n-1}$, there is also a Nash equilibrium on the longest path. There are no other Nash equilibria, for any value of r.*

Proof. We first quickly show that $r \geq 2\varepsilon$ guarantees an optimal Nash equilibrium. Suppose A_2 takes the optimal path P_0. While standing at s, A_1 evaluates the cost to t as $b - r/2$ and the cost of v_1 as c; so with $r \geq 2\varepsilon$, A_1 (weakly) prefers to go directly to t. It's easy to see, using a similar calculation, that when $r \leq 2\varepsilon c^{n-1}$, there is a Nash equilibrium on the longest path, P_n.

To show that there are no other Nash equilibria, suppose A_2 takes P_i, where $1 \leq i < n$. Note that A_1 gets the same **reward** from paths P_1 to P_{i-1} and the same reward from paths P_{i+1} to P_n. Since $b > c$, the agent will always procrastinate to at least v_{i-1}, and if he chooses to go past v_i, he will procrastinate until v_n. Now, suppose A_1 is standing at v_{i-1}. He will take path P_{i-1} if:

$$\begin{aligned} bc^{i-1} - r &< c^i - r/2 \\ (c+\varepsilon)c^{i-1} - c^i &< r/2 \\ r &> 2\varepsilon c^{i-1} \end{aligned}$$

Either the reward satisfies this, which shows that P_i is not a Nash equilibrium, or $r \leq 2\varepsilon c^{i-1}$, and A_1 (may) continue to v_i (depending on tie-breaking). Suppose he continues to v_i. Then, he (weakly) prefers P_i if:

$$\begin{aligned} bc^i - r/2 &\le c^{i-1} \\ r &\ge 2\varepsilon c^i \end{aligned}$$

However, $r \le 2\varepsilon c^{i-1} < 2\varepsilon c^i$, so r does not satisfy this. Thus, A_1 prefers to procrastinate until v_n. In summary, if A_2 takes path P_i, A_1 either prefers P_{i-1} or P_n (depending on the reward). The same argument applies to A_2, so there are no Nash equilibria on paths $P_1, \ldots, P_{n-1}$, regardless of r. □

The ability to enable optimal behavior with just a constant reward stands in striking contrast to the setting where agents can abandon the task at any time. With neither competition nor internal rewards, the designer would have to place an exponentially high reward on t (i.e. $r \ge c^n$) for biased agents to finish the task.[2]

We also discuss two properties of this problem which suggest that encouraging better behavior is somewhat "robust" to several types of uncertainty that the task designer or agents may face.

1. As the path A_2 takes is closer to optimal, the reward required to motivate A_1 to "good" behavior decreases, where good behavior is when an agent better optimizes their utility by taking a shorter path. Specifically, if A_1 believes A_2 will take P_i, $r > 2\varepsilon c^{i-1}$ convinces A_1 to take P_{i-1} (rather than procrastinating to P_n).

2. As the reward r increases, if c is held fixed, then agents with a wider range of biases b will find the optimal path a Nash equilibrium. On the other hand, if b is held fixed, then as r increases the cost ratio c^n decreases.

The proofs of these properties follow directly from the proof of the above theorem.

We now informally discuss how these properties bear on several types of uncertainty. First, suppose the mechanism designer knows the bias parameter b and cost parameter c, but the agents are uncertain about how the other agent will behave. If the designer sets r too close to 2ε and A_1 believes that A_2 will behave even slightly sub-optimally (e.g. A_1 believes A_2 will take P_3), then A_1 will procrastinate all the way to P_n. If $r \ge 2\varepsilon c^2$, A_1 would have taken P_2 instead, for an exponentially lower cost. The designer can leverage property 1 in this situation. For one example, an additional reward of r^* could be (publicly) offered to only A_2 (if they win/half if tie) to encourage optimal behavior. This would make A_1 more confident that A_2 will take the optimal path, and thus the reward r required to motivate A_1 to behave optimally would be lower as well. If we were to extend our model to n uncertain agents, as long as one agent is likely to behave optimally, property 1 says that it will be less expensive to convince the other agents to behave optimally. The same idea is helpful when the mechanism designer is uncertain about the biases of the agents (in addition to the agents' uncertainty); the agents need only think that the least biased agent among them will act optimally. If this is likely the case, the rest will be easier to convince.

Finally, in addition to bias uncertainty, suppose the mechanism designer is uncertain about c. The second property suggests that increasing the reward is doubly beneficial here. The obvious benefit is that more agents will find a Nash equilibrium for the optimal path. The more subtle benefit that if the agents don't think P_0 is a Nash equilibrium, either (i) they have very high bias (in which case, we should expect a high reward to be necessary), or (ii) c turned out to be very low,

[2] However, with internal *edge* rewards, the designer could spend the same total reward to motivate two agents by directly adding a reward to the (s, t) edge. This equivalence doesn't hold in for general graphs, as we show in the next section.

and so the worst-case behavior of c^n is not as bad. Said more simply, as the designer increases the reward, they simultaneously attract more agents towards optimal behavior and lower the "badness" of suboptimal behavior (where the badness is with respect to their uncertainty over c).[3]

3.2 Nash Equilibria in General Graphs

We now generalize to arbitrary graphs, but we first make an important assumption. To focus exclusively on the irrationality as opposed to the optimization problem between cheap, long paths and short, expensive paths, we assume that the cheapest path is also the *uniquely* quickest path – i.e., the optimal path without rewards is also the path where one is most likely to win. This is the case in the n-fan. In this setting, the problem is trivial for unbiased agents; simply take the cheapest/quickest path (it's the only non-dominated path). But for biased agents, the problem is still interesting. We now show that a small amount of competition/reward suffices for a Nash equilibrium on the optimal path.

Theorem 2. *Suppose G is an arbitrary task graph with optimal/cheapest path O. Then, a reward of $r \geq 2b \cdot \max_{e \in O} c(e)$ guarantees a Nash equilibrium on the optimal path, for two agents with bias b.*

Proof. Recall from earlier that a biased agent picks a successor node after having travelled path $P(s \to v)$ via:

$$S_n(u) = \operatorname*{argmin}_{v:(u,v)\in E} b \cdot c(u,v) + \min_{P(v\to t)\in\mathcal{P}(v\to t)} c(P(v \to t)) - \mathbb{E}[R(P(s \to u) \cup (u,v) \cup P(v \to t))]$$

We know that for any vertex v^* on the optimal path, the path that minimizes the second term is just the fragment of the optimal path from $v^* \to t$ (it is both the quickest and cheapest way to get from v^* to t). Further, any deviation from the optimal path results in no reward (since, to find a Nash equilibrium on the optimal path, we assume the other agent takes the optimal path). So, for any v not on the optimal path, the expected reward is always zero, and thus the path that minimizes the second term is again the cheapest path from $v \to t$. Thus, our successor equation simplifies to:

$$S_n(u) = \operatorname*{argmin}_{v:(u,v)\in E} b \cdot c(u,v) + C_o(v) - r/2 \cdot \mathbf{1}\{D\}$$

Where $C_o(v)$ denotes the optimal cost from v to t and $\mathbf{1}\{D\}$ is simply an indicator variable that's 1 if the agent has not deviated from the optimal path.

Now, let $O = (s = v_0^*, v_1^*, v_2^*, \ldots, t = v_l^*)$ be the optimal path and suppose A_2 takes this path. In order for A_1 to stay on O, we require, for all i:

$$\begin{aligned}
&S_n(v_i^*) = v_{i+1}^* \\
\iff &v_{i+1}^* = \operatorname*{argmin}_{v:(v_i^*,v)\in E} b \cdot c(v_i^*, v) + C_o(v) - r/2 \cdot \mathbf{1}\{D\} \\
\iff &\forall v : (v_i^*, v) \in E, bc(v_i^*, v) + C_o(v) > bc(v_i^*, v_{i+1}^*) + C_o(v_{i+1}^*) - r/2
\end{aligned}$$

Now, let i be arbitrary, let $v \neq v_i^*$ be an arbitrary neighbor of v_i^*, and for ease of notation, let $c = c(v_i^*, v), c' = c(v_i^*, v_{i+1}^*), C_o = C_o(v)$, and $C_o' = C_o(v_{i+1}^*)$. Then, we get the following bound on the reward:

$$r/2 > b(c' - c) + C_o' - C_o$$

[3] Note that property 2 does not address asymptotic behavior, as c^n will still be exponentially large. The property is more relevant when b and c are significant constants, and n is not too large.

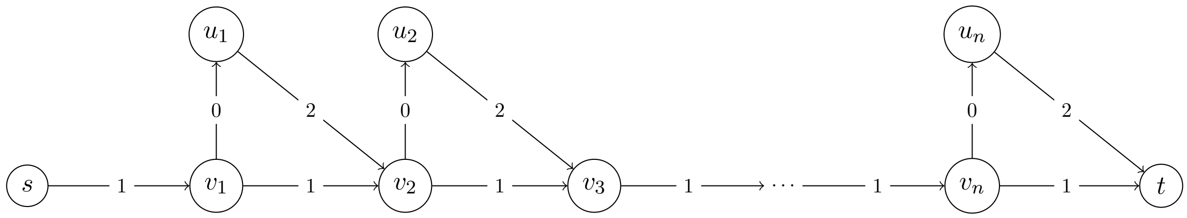


Figure 6: A graph with many suboptimal deviations. For an agent with bias $b > 2$, a designer with edge rewards must spend $O(n)$ total reward for optimal behavior ($b - 2$ on each v edge). In our competitive setting, only $2(b - 2)$ total reward is required.

To get a rough sufficient bound, we leverage optimality to show that $bc' > b(c' - c) + C'_o - C_o$:

$$
\begin{aligned}
c + C_o &\geq c' + C'_o && \text{(By the optimality of path } O\text{)} \\
c + C_o &\geq C'_o \\
bc + C_o &> C'_o \\
bc' &> bc' - bc + C'_o - C_o && \text{(Arithmetic)}
\end{aligned}
$$

Thus, it suffices to set $r > 2bc'$ in order to ensure $S_n(v_i^*) = v_{i+1}^*$. Repeating this argument for all i, we see that a sufficient reward is $r = 2b \cdot \max_{e \in O} c(e)$, and thus a constant amount of reward guarantees a Nash equilibrium on the optimal path. □

For a point of comparison, even internal edge rewards (which required the same reward budget as competitive rewards for the n-fan) require at least $O(n)$ total reward in some instances. For a concrete example, see Figure 6.

However, the theorem bound is not particularly tight. So, for completeness, we present Algorithm 1, a poly time algorithm that takes in a graph G, path Q, and bias b, and determines whether it's possible to get a Nash equilibrium on Q, and if so, the minimum required reward. It may not be possible to construct a Nash equilibrium on an arbitrary path. In particular, define the *natural* path an agent with bias b takes as the path the agent would take in the absence of competition. In our model, increasing the reward can only ever convince the agents to take quicker paths. One consequence is that we can never convince agents to take paths slower than their natural path. Further, if the agent would deviate from Q onto a quicker or tied path, no reward would convince them otherwise. The main challenge is in efficiently computing $\min_{P(v \to t) \in \mathcal{P}(v \to t)} c(P(v \to t)) - \mathbb{E}[R(P(s \to u) \cup (u, v) \cup P(v \to t))]$. However, in the case of a Nash equilibrium where A_1 knows A_2 will take Q, it suffices to compute the cheapest path, the cheapest path where A_1 ties A_2, and the cheapest path where A_1 beats A_2. These paths may not be distinct, and they may not all exist, but the minimizer must be among these. We now present the algorithm.

Algorithm 1: Set Nash equilibrium on given path, if possible. Uses $\mathsf{Bellman\text{-}Ford}(v, i)$ as a subroutine, to get the cost of the cheapest $v \to t$ path with at most i edges.

```
Input: A DAG G, with source s and sink t, path Q in G, and bias factor b
Output: The minimum reward r for Q to be a Nash equilibrium, or ⊥ if not possible.
k ← len(Q)
r ← 0
foreach (u, v) ∈ Q do
    c1, c2, c3 ← Bellman-Ford(v, ∞), Bellman-Ford(v, k), Bellman-Ford(v, k − 1)
    c* ← min(c1, c2 − r/2, c3 − r)
    foreach (u, v' ≠ v) ∈ G do
        d1, d2, d3 ← Bellman-Ford(v', ∞), Bellman-Ford(v', k), Bellman-Ford(v', k − 1)
        d* ← min(d1, d2 − r/2, d3 − r)
        if b · c(u, v') + d* < b · c(u, v) + c* then
            if d* = d2 − r/2 or d* = d3 − r then
                return ⊥
            else
                r ← max(r, 2b(c(u, v) − c(u, v')) + 2(c* − d*))
    k ← k − 1
return r
```

4 Bayes-Nash Equilibria with Bias Uncertainty

One of the shortcomings of the model studied thus far is that agents are assumed to have publicly known, identical biases. This presents both general problems and problems specific to the mechanism design perspective. In general, it seems odd for the agents to know the exact present bias of their opponent, or for the agents to always have the same bias. And from a design perspective, the task designer might not have full knowledge of the agents' biases, and so may be unsure how to set the reward. We thus add a layer of *bias uncertainty* to our model. The agents' biases are represented by random variables B_1 and B_2 drawn iid from distribution B, which is publicly known to both the agents and the designer. b_1 and b_2 correspond to the realizations of these random variables. Our goal is now to construct, as cheaply as possible, Bayes-Nash equilibria (BNE) where agents behave optimally with high probability. In this section, we only have concrete results on the n-fan; we start by explaining these and then discuss why generalizing to arbitrary graphs is difficult.

4.1 Bayes-Nash Equilibria on the n-fan

As before, let P_i represent the path that includes edge (v_i, t), and P_0 represent the optimal path. Let $\Pr[A_2 \to P]$ denote the probability that A_2 takes path P, or any path in P, when P is a set. The probability is over $B_2 \sim B$. Let $P_{>i}$ denote the set of paths $\{P_{i+1}, \ldots, P_n\}$. We start with a basic claim:

Claim 1. *When standing at vertex v_i, A_1 prefers P_i over P_{i+1} if:*

$$r/2 \Pr[A_2 \to \{P_i, P_{i+1}\}] > c^i(b_1 - c)$$

Proof. This is just a basic calculation:

$$
\begin{aligned}
r \cdot \Pr[A_2 \to P_{>i}] + r/2 \cdot \Pr[A_2 \to P_i] - b_1 c^i &> r \cdot \Pr[A_2 \to P_{>i+1}] + r/2 \cdot \Pr[A_2 \to P_{i+1}] - c^{i+1} \\
r(\Pr[A_2 \to P_{>i}] - \Pr[A_2 \to P_{>i+1}]) + r/2(\Pr[A_2 \to P_i] - \Pr[A_2 \to P_{i+1}]) &> c^i(b_1 - c) \\
r(\Pr[A_2 \to P_{i+1}]) + r/2(\Pr[A_2 \to P_i] - \Pr[A_2 \to P_{i+1}]) &> c^i(b_1 - c) \\
r/2 \Pr[A_2 \to \{P_i, P_{i+1}\}] &> c^i(b_1 - c)
\end{aligned}
$$

Where the left hand side is A_1's expected utility from going immediately to t and the right hand side is their perceived utility from procrastinating. □

This claim matches intuition; due to the tying mechanism, A_1 gets a reward $r/2$ higher by going to t from v_i if A_2 either takes P_i or P_{i+1}. Note that A_1 is comparing P_i to P_{i+1} because the latter describes what he believes he will do if he procrastinates. We now describe a Bayes-Nash equilibrium in the n-fan.

Theorem 3. *Let G be an n-fan with reward r and suppose B_1, B_2 are drawn from distribution B with CDF F. Let p be the solution to $F(\frac{rp}{2} + c) = p$. If $p > \frac{1}{c^{n-1}+1}$, then the following strategy is a Bayes-Nash equilibrium:*

$$
P(b) = \begin{cases} \textit{take } P_0, & b \le \frac{rp}{2} + c \\ \textit{take } P_n, & \textit{otherwise} \end{cases}
$$

Further, in this equilibrium, both agents will take P_0 with probability p. So the expected cost ratio will be $p + (1-p)c^n$.

Proof. Using Claim 1, at s, A_1 prefers P_0 over P_1 if:

$$
\begin{aligned}
r/2 \Pr[A_2 \to \{P_0, P_1\}] &> b_1 - c \\
b_1 &< \frac{r \Pr[A_2 \to P_0]}{2} + c \qquad \text{Since } \Pr[A_2 \to P_1] = 0
\end{aligned}
$$

Let $\Pr[A_2 \to P_0] = p$. Then, for a symmetric BNE, we want:

$$
\begin{aligned}
\Pr[A_1 \to P_0] &= \Pr[A_2 \to P_0] \\
\Pr[B_1 \le \frac{rp}{2} + c] &= p \\
F\left(\frac{rp}{2} + c\right) &= p
\end{aligned}
$$

So if p satisfies this equation, by Claim 1, A_1 prefers to go from s to t directly. Now, consider the case where $b_1 > \frac{rp}{2} + c$. Then, A_1 prefers to go to v_1. Again using Claim 1, v_1, A_1 will prefer t over v_2 if:

$$
\begin{aligned}
r/2 \Pr[A_2 \to \{P_1, P_2\}] &> c(b_1 - c) \\
0 &> b_1 - c
\end{aligned}
$$

Since we know that $b_1 > \frac{rp}{2} + c$, it's certainly the case that $b_1 > c$, so this never holds. Thus, A_1 prefers to go to v_2. The same argument can be repeated until the agent reaches v_{n-1}. Then, the agent will prefer t over v_n if:

$$
\begin{aligned}
r/2 \Pr[A_2 \to \{P_{n-1}, P_n\}] &> c^{n-1}(b_1 - c) \\
r/2(1-p) &> c^{n-1}(b_1 - c) \\
b_1 &< \frac{r(1-p)}{2c^{n-1}} + c
\end{aligned}
$$

Combining our assumption that $b_1 > \frac{rp}{2} + c$ with this inequality:

$$\begin{aligned}
\frac{rp}{2} + c < b_1 &< \frac{r(1-p)}{2c^{n-1}} + c \\
\frac{rp}{2} + c &< \frac{r(1-p)}{2c^{n-1}} + c \\
p &< \frac{1-p}{c^{n-1}} \\
p &< \frac{1}{c^{n-1}+1}
\end{aligned}$$

Thus, if $p > \frac{1}{c^{n-1}+1}$, the agent will always prefer P_n when their bias $b_1 > \frac{rp}{2} + c$. Since B_1 and B_2 are both drawn iid from B, by symmetry neither agent has an incentive to deviate from the strategy, showing that it is a Bayes-Nash equilibrium. □

Note that while it's possible (for some distributions B and very low rewards r) for the solution to $p = \Pr[B_1 \le \frac{rp}{2} + c]$ to be less than $\frac{1}{c^{n-1}+1}$, we mostly ignore this case; the distributions we explore won't behave this way.

We now use the theorem to understand how much reward is required for optimal behavior with high probability, or a low expected cost ratio (which is a much stronger requirement). We first make some general remarks and then look at specific distributions.

Since the expected cost is $p + c^n(1-p)$, in order for this to be low, $1-p$ has to be close to $1/c^n$. Plugging this in to the CDF:

$$F\left(\frac{r}{2}\left(1 - \frac{1}{c^n}\right) + c\right) = 1 - \frac{1}{c^n}$$

Which essentially requires that exponentially little probability mass remains after $r/2$ distance from c. So, with an exponential distribution, this requires linear distance/reward, and with a heavier tailed distribution like the Equal Revenue distribution, this requires exponential distance. But we may be content with simply guaranteeing optimal behavior with high probability. So, we describe more precisely the relationship between r and p for several distributions.

Lemma 1. *Suppose B_1 and B_2 are drawn from an equal revenue distribution that's shifted over by c (so $b_1, b_2 > c$). Then, for any $r \ge 2$, the probability of optimal behavior in the BNE from Theorem 3 is $1 - \frac{2}{r}$.*

Proof. See Appendix A.1. □

Because this distribution is heavy tailed, the expected cost remains exponential unless r is exponential. However, if r is any increasing function of n, then w.h.p. both agents will take the optimal path. Now, we look at the exponential distribution.

Lemma 2. *Suppose B_1 and B_2 are drawn from an exponential distribution, $\mathbf{Exp}(\lambda)$, that's shifted over by c (so $b_1, b_2 > c$). Then, for any r, the probability of optimal behavior in the BNE from Theorem 3 is at least $1 - \frac{1}{\exp(\lambda r/2)}$.*

Proof. See Appendix A.2. □

This means that if $r = O(n)$, we get a constant expected cost ratio (and extremely high probability of optimal behavior). We finally consider the simple example of a uniform distribution.

Lemma 3. *Suppose B_1 and B_2 are drawn from the uniform distribution, $U[c,d]$ (so $b_1, b_2 > c$). Then, for $r \geq 2(d-c)$, the probability of optimal behavior in the BNE from Theorem 3 is 1. For all $r < 2(d-c)$, this probability is zero.*

Proof. See Appendix A.3. □

More generally, for any distribution bounded by b^*, computing the reward that guarantees a Nash equilibrium if both agents had public biases b^* also suffices to guarantee a BNE in this setting (and in particular, this BNE always results in optimal behavior). For the case of the uniform distribution, this is only reward that ever results in optimal behavior. As a direct consequence of this lemma, the expected cost ratio is 1 with the reward $r = O(d)$.

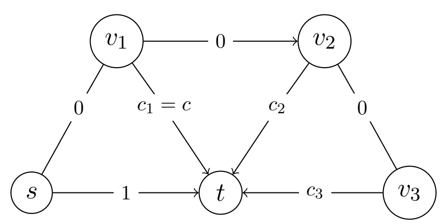
4.2 Difficulties Finding Bayes-Nash Equilibria in Arbitrary Graphs

The simplest way to generalize our BNE from the n-fan would be to consider the following schematic:

$$S_y(b) = \begin{cases} \text{take } O, & b \leq y \\ \text{take } P, \text{ for some fixed } P & \text{otherwise} \end{cases}$$

Where y ideally would depend on only the costs in the graph, the reward r, and p, the solution to $F(y) = p$. So the agents would either take the optimal path, if their bias was sufficiently small, or some path P that can "catch" agents with high bias. In the fan graph, this was the path around the fan. That path happened to be the natural path for any bias, and perhaps more importantly, the "limiting" natural path as $b \to \infty$ (i.e. the greedy/myopic path, where you simply take the lowest cost edge at each step). But neither of these ideas generalize; this simple schematic will not produce a BNE in an arbitrary graph, as the following counter example shows.

Consider the modified 3-fan below. Let $1 < c_1 = c < c^2 < c_2 < c_2^2 < c_3$. So the cost of the edges to t increase faster than doubling.



We claim the following:

Claim 2. *There is no symmetric BNE for this graph that assigns positive probability to only two paths, assuming the bias is drawn from a distribution with support on $[1, \infty]$.*

Proof. See Appendix A.4. □

The main idea behind the proof is to notice first that path P_0 and P_3 will always be taken, when the bias is very low or very high. We then define the interval of biases (the interval depends on the reward, costs, and probability of optimal behavior) wherein A_1 wants to take P_1 or P_2 and argue that at least one of the intervals must be non-empty. Thus, the main challenge with finding a BNE is finding a set of paths $\{P_i\}$ and bias intervals $\{I_i\}$ such that, for all intervals I_i, A_1 must want to traverse P_i if their bias falls in I_i. The complication is that A_1's decision to traverse P_i depends on the distribution over A_2's choice of path, which is induced by the intervals I_i and the bias distribution. These interweaving constraints are difficult to manage even for simple graphs, as preliminary work on the modified n-fan suggests.

5 Conclusion and Future Directions

We investigated the impact of competition on present bias in competitive task-completion problems, in a powerful graph-theoretic model. In instances that that are trivial for unbiased agents, biased agents can experience exponentially higher cost. However, when there is no uncertainty in the model, we found that just a small amount of competition (and reward) could divert present-biased agents from their exponential worst case behavior to the optimal path. When the biases of the agents become uncertain, the problem grows much more complex. We were able to characterize a simple Bayes-Nash equilibrium for the n-fan, a graph that encodes the fundamental danger of present bias. If our goal is to ensure that the agents behave optimally with high probability, we found that for a wide range of distributions, offering reward that's any increasing function of n suffices. However, the harms of suboptimal behavior in this graph are extreme. So, for the much stronger goal of ensuring low expected cost ratio, we found that, for distributions that are not heavy-tailed, it suffices to offer reward linear in n.

This thesis is a first step at painting a more optimistic picture than much of the work surrounding present bias. Our results highlight why, in naturally competitive settings, otherwise biased agents might behave optimally. Further, task/mechanism designers can use our results to directly alleviate the harms of present bias. This competitive model is both a more natural model than other motivation schemes, such as internal edge rewards, and can more cheaply ensure optimal behavior.

However, our results are not comprehensive. In particular, we require significant assumptions about the task graphs, and our strongest results hold only with no uncertainty in the model. So one obvious future direction is to make our results more general by dropping these assumptions. Specifically, can we drop the assumption that the cheapest path is the quickest, and instead build upon our unbiased competitive results? One direct goal would be to mathematically characterize which paths can be made a Nash equilibrium with biased agents, rather than our current algorithmic approach. Any patterns in such paths may be helpful for constructing a Bayes-Nash equilibrium in an arbitrary graph, which is the other obvious direction.

We also offer two more substantive, but speculative, directions. The first is to somehow amplify the amount of competition in the model. Currently, there are simply two agents competing for a reward. Will we be able to encourage optimal behavior more cheaply when n agents are competing? We conjecture that this is true when bias uncertainty is in effect; the agents need to be concerned only with the *most* optimal agent, and should behave more optimally in response.

Lastly, we could extend our work beyond cost ratios, moving to the model where agents can abandon their path at any point. Our bounds on the reward required for optimal behavior in equilibrium should translate to the reward required for agents to complete the task. For one, this would allow us to integrate results on *sunk-cost* bias, represented as an intrinsic cost for abandoning a task that's proportional to the amount of effort expended. [KOR17] show that agents who are sophisticated with regard to their present-bias, but naive with respect to their sunk cost bias can experience exponentially high cost before abandoning their traversal (this is especially interesting because sophisticated agents without sunk cost bias behave nearly optimally). Can competition alleviate this exponential worst case? This would also be a natural way to investigate sophisticated agents in competitive settings.

References

[AK19] Susanne Albers and Dennis Kraft. Motivating time-inconsistent agents: A computational approach. *Theory of computing systems*, 63(3):466–487, 2019.

[KO14] Jon Kleinberg and Sigal Oren. Time-inconsistent planning: a computational problem in behavioral economics. In *Proceedings of the fifteenth ACM conference on Economics and computation*, pages 547–564, 2014.

[KOR16] Jon Kleinberg, Sigal Oren, and Manish Raghavan. Planning problems for sophisticated agents with present bias. In *Proceedings of the 2016 ACM Conference on Economics and Computation*, pages 343–360, 2016.

[KOR17] Jon Kleinberg, Sigal Oren, and Manish Raghavan. Planning with multiple biases. In *Proceedings of the 2017 ACM Conference on Economics and Computation*, pages 567–584, 2017.

[TTW+17] Pingzhong Tang, Yifeng Teng, Zihe Wang, Shenke Xiao, and Yichong Xu. Computational issues in time-inconsistent planning. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

A Proofs

A.1 Proof of Lemma 1

Proof. The equal revenue distribution described has CDF $F(z) = 1 - \frac{1}{z-c+1}$. Now, applying Theorem 3:

$$\begin{aligned} 1 - \frac{1}{\frac{rp}{2} + c - c + 1} &= p \\ \frac{rp}{2} + 1 - 1 &= p\left(\frac{rp}{2} + 1\right) \\ \left(\frac{r}{2}\right)p^2 + \left(1 - \frac{r}{2}\right)p &= 0 \\ p &= 0 \text{ or } \frac{r-2}{r} \end{aligned}$$

So with reward $r \geq 2$, $p = \frac{r-2}{r}$. □

A.2 Proof of Lemma 2

Proof. We first solve for p in Theorem 3:

$$\begin{aligned} \Pr[B_1 \leq \frac{rp}{2} + c] &= p \\ 1 - e^{-\lambda(\frac{rp}{2} + c - c)} &= p \\ 1 - e^{-\frac{\lambda rp}{2}} &= p \\ \frac{\ln(1-p)}{p} &= \frac{-\lambda r}{2} \\ p &= 0 \text{ or } 1 + \frac{2W(-\lambda r/2 \cdot e^{-\lambda r/2})}{\lambda r} \end{aligned}$$

Where W refers to the Lambert W function, i.e. $W(z) = x \iff xe^x = z$. The Taylor series of W around 0 is:

$$W(x) = \sum_{n=1}^{\infty} \frac{(-n)^{n-1}}{n!} x^n = x - x^2 + \frac{3}{2}x^3 - \dots$$

For our application, notice that $x = -\lambda r/2 \cdot e^{-\lambda r/2}$ is negative; thus, $W(x) \le x$ as the even powers are subtracted and odd powers added. Since this value is also close to 0 for somewhat large $r\lambda$, this inequality isn't too inexact. So:

$$p \ge 1 + \frac{2}{\lambda r} \cdot \frac{-\lambda r}{2e^{\lambda r/2}} = 1 - \frac{1}{e^{\lambda r/2}}$$

□

A.3 Proof of Lemma 3

Proof. For the first part of the lemma, note that if $rp/2 + c \ge d$, then $F(rp/2 + c) = 1$, so $p = 1$ holds if $r/2 + c \ge d$, or equivalently, $r \ge 2(d - c)$. If we assume $rp/2 + c < d$, then using CDF $F(z) = \frac{z-c}{d-c}$ to solve for p:

$$\begin{aligned} \frac{\frac{rp}{2} + c - c}{d - c} &= p \\ \frac{r}{2}p &= (d - c)p \end{aligned}$$

Since $r < 2(d - c)$, this is satisfied only with $p = 0$. □

A.4 Proof of Claim 2

Proof. Note that as $b := b_1 \to \infty$, for any fixed reward, the agent will always take the greedy option of choosing the cheapest edge at every step. This will cause them to take the path P_3. Further if their bias is below c_1, they will take the optimal path, P_0. So, any BNE must assign positive probability to those two paths. We now prove, by contradiction, that it must assign positive probability to either P_1 or P_2.

Let p be $\Pr[A_2 \to P_0]$ and $1 - p = \Pr[A_2 \to P_3]$. At s, A_1 prefers to procrastinate to v_1 when $b > c + rp/2$, so suppose this is the case. At v_1, A_1 would take P_1 if:

$$\begin{aligned} bc - r(1-p) &< c_2 - r(1-p) \\ b &< c_2/c \end{aligned}$$

So, if $b \in [c + rp/2, c_2/c]$, A_1 would take P_1. If this interval is non-empty, then we're done. If it's not, then:

$$\begin{aligned} c_2 c &> c + rp/2 \\ rp &> \frac{2c_2}{c} - 2c \\ r &> \frac{2c_2 - 2c^2}{cp} \end{aligned}$$

If the reward is at least this high, then A_1 always goes to v_2 at v_1. Now, at v_2 (where we know $b > c_2/c$), A_1 takes P_2 when:

$$\begin{aligned} bc_2 - r(1-p) &< c_3 - \frac{r}{2}(1-p) \\ bc_2 &< c_3 + \frac{r}{2}(1-p) \\ b &< \frac{2c_3 + r(1-p)}{2c_2} \end{aligned}$$

As before, if $b \in [c_2/c, \frac{2c_3+r(1-p)}{2c_2}]$, A_1 would take P_2. We claim that this interval must be non-empty. Suppose it wasn't, and $c_2/c > \frac{2c_3+r(1-p)}{2c_2}$. Then:

$$\begin{aligned} r(1-p) + 2c_3 &< 2c_2^2/c \\ r &< \frac{2c_2^2 - 2c_3c}{c(1-p)} \end{aligned}$$

However, this upper bound causes a contradiction – from the definitions, we know that $c_2^2 < c_3$, so clearly $c_2^2 < c_3c$, and thus this bound requires $r < 0$. This contradicts our typical requirement that $r \geq 0$, but even ignoring this requirement, this also contradicts our earlier bound that $r > \frac{2c_2 - 2c^2}{cp}$.

Our proof thus says that A_1 will either take P_1 or P_2 with positive probability. $\square$